PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

Search Results

Search Results for: [instruction<AND>((register allocation))]
Found 969 of 105,778 searched. → Rerun within the Portal

Warning: Maximum result set of 200 exceeded. Consider refining.

Search within Results

GO > Advanced Search > Search Help/Tips

Sort by: Title Publication Publication Date Score

Results 1 - 20 of 200 **short listing**

Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page

1 Register Allocation for Banked Register File 100%
Jinpyo Park , Je-Hyung Lee , Soo-Mook Moon
ACM SIGPLAN Notices August 2001
Volume 36 Issue 8

A banked register file is a register file partitioned into banks. A register in a banked register file is addressed with the register number in conjunction with the active bank number. A banked register file may be employed to reduce the number of bits for register operands in the instruction encoding at the cost of bank changes and inter-bank data transfers. Although a banked register file is introduced to provide sufficient registers and reduce memory traffic, it may on the other hand infla ...

2 Demand-driven register allocation 100%
Todd A. Proebsting , Charles N. Fischer
ACM Transactions on Programming Languages and Systems (TOPLAS) November 1996
Volume 18 Issue 6

> A new global register allocation technique, demand-driven register allocation, is described. Demand-driven register allocation quantifies the costs and benefits of allocating variables to registers over live ranges so that high-quality allocations can be made. Local allocation is done first, and then global allocation is done iteratively beginning in the most deeply nested loops. Because local allocation precedes global allocation, demand-driven allocation does not interfer ...

3 Experience with a software-defined machine architecture 100%
David W. Wall
ACM Transactions on Programming Languages and Systems (TOPLAS) May 1992
Volume 14 Issue 3

> We have built a system in which the compiler back end and the linker work together to present an abstract machine at a considerably higher level than the actual machine. The intermediate language translated by the back end is the target language of all high-level compilers and is also the only assembly language generally available. This lets us do intermodule register allocation, which would be harder if some of the code in the program had come from a traditional assembler, out of sight of ...

4 Code generation of nested loops for DSP processors with heterogeneous registers and structural pipelining 100%
Wei-Kai Cheng , Youn-Long Lin
ACM Transactions on Design Automation of Electronic Systems (TODAES) July 1999
Volume 4 Issue 3

> We propose a microcode-optimizing method targeting a programmable DSP processor. Efficient generation of microcodes is essential to better utilize the computation power of a DSP processor. Since most state-of-the-art DSP processors feature some sort of irregular architectures and most DSP applications have nested loop constructs, their code generation is a nontrivial task. In this paper, we consider two features frequently found in contemporary DSP processors &mdash; structural pipelining a ...

5 An experimental study of several cooperative register allocation and instruction scheduling strategies 100%
Cindy Norris , Lori L. Pollock
Proceedings of the 28th annual international symposium on Microarchitecture December 1995

6 A register allocation technique using guarded PDG 100%
Akira Koseki , Hideaki Komatsu , Yoshiaki Fukazawa
Proceedings of the 10th international conference on Supercomputing January 1996

7 tcc: a system for fast, flexible, and high-level dynamic code generation 100%
Massimiliano Poletto , Dawson R. Engler , M. Frans Kaashoek
ACM SIGPLAN Notices , Proceedings of the 1997 ACM SIGPLAN conference on Programming language design and implementation May 1997
Volume 32 Issue 5

8 Code scheduling and register allocation in large basic blocks 100%
J. R. Goodman , W.-C. Hsu
Proceedings of the 2nd international conference on Supercomputing June 1988

> We discuss the issues about the interdependency between code scheduling and register allocation. We present two methods as solutions: (1) an integrated code scheduling technique; and (2) a DAG-driven register allocator. The integrated code scheduling method combines two scheduling techniques&mdash;one to reduce pipeline delays and the other to minimize register usage&mdash;into a single phase. By keeping track of the number of available registers, the scheduler can choose the appropriate sc ...

9 Precise register allocation for irregular architectures 100%
Timothy Kong , Kent D. Wilken
Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture November 1998

10 Register allocation with instruction scheduling 100%
Shlomit S. Pinter
ACM SIGPLAN Notices , Proceedings of the conference on Programming language design and implementation June 1993
Volume 28 Issue 6

We present a new framework in which considerations of both register allocation and instruction scheduling can be applied uniformly and simultaneously. In this framework an optimal coloring of a graph, called the parallel interference graph, provides an optimal register allocation and preserves the property that no false dependences are introduced, thus all the options for parallelism are kept for the scheduler to handle. For this framework we provide heuristics for trading ...

11 Integrating register allocation and instruction scheduling for RISCs 100%
David G. Bradlee , Susan J. Eggers , Robert R. Henry
ACM SIGARCH Computer Architecture News , Proceedings of the fourth international conference on Architectural support for programming languages and operating systems April 1991
Volume 19 Issue 2

12 Improvements to graph coloring register allocation 100%
Preston Briggs , Keith D. Cooper , Linda Torczon
ACM Transactions on Programming Languages and Systems (TOPLAS) May 1994
Volume 16 Issue 3

We describe two improvements to Chaitin-style graph coloring register allocators. The first, optimistic coloring, uses a stronger heuristic to find a k-coloring for the interference graph. The second extends Chaitin's treatment of rematerialization to handle a larger class of values. These techniques are complementary. Optimistic coloring decreases the number of procedures that require spill code and reduces the amount of spill code when sp ...

13 Fast, effective code generation in a just-in-time Java compiler 100%
Ali-Reza Adl-Tabatabai , Micha? Cierniak , Guei-Yuan Lueh , Vishesh M. Parikh , James M. Stichnoth
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN '98 conference on Programming language design and implementation May 1998
Volume 33 Issue 5

14 C and tcc: a language and compiler for dynamic code generation 99%
Massimiliano Poletto , Wilson C. Hsieh , Dawson R. Engler , M. Frans Kaashoek
ACM Transactions on Programming Languages and Systems

(TOPLAS) March 1999
Volume 21 Issue 2

Dynamic code generation allows programmers to use run-time information in order to achieve performance and expressiveness superior to those of static code. The 'C(Tick C) language is a superset of ANSI C that supports efficient and high-level use of dynamic code generation. 'C provides dynamic code generation at the level of C expressions and statements and supports the composition of dynamic code at run time. These features enable programmers to add dynamic code generation ...

15 Code reuse in an optimizing compiler 99%
Ali-Reza Adl-Tabatabai , Thomas Gross , Guei-Yuan Lueh
ACM SIGPLAN Notices , Proceedings of the eleventh annual conference on Object-oriented programming systems, languages, and applications October 1996
Volume 31 Issue 10

16 Quality and speed in linear-scan register allocation 99%
Omri Traub , Glenn Holloway , Michael D. Smith
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN '98 conference on Programming language design and implementation May 1998
Volume 33 Issue 5

17 A schedular-sensitive global register allocator 99%
C. Norris , L. L. Pollock
Proceedings of the 1993 ACM/IEEE conference on Supercomputing December 1993

18 A Dynamic Programming Approach to Optimal Integrated Code Generation 99%
Christoph Keßler , Andrzej Bednarski
ACM SIGPLAN Notices August 2001
Volume 36 Issue 8

Phase-decoupled methods for code generation are the state of the art in compilers for standard processors but generally produce code of poor quality for irregular target architectures such as many DSPs. In that case, the generation of efficient code requires the simultaneous solution of the main subproblems instruction selection, instruction scheduling, and register allocation, as an integrated optimization problem.

In contrast to compilers for standard processors, code generation for ...

19 A simple interprocedural register allocation algorithm and its effectiveness for LISP 99%

Peter A. Steenkiste , John L. Hennessy
ACM Transactions on Programming Languages and Systems (TOPLAS) January 1989
Volume 11 Issue 1

Register allocation is an important optimization in many compilers, but with per-procedure register allocation, it is often not possible to make good use of a large register set. Procedure calls limit the improvement from global register allocation, since they force variables allocated to registers to be saved and restored. This limitation is more pronounced in LISP programs due to the higher frequency of procedure calls. An interprocedural register allocation algorithm is developed by simp ...

20 The priority-based coloring approach to register allocation 99%

Fred C. Chow , John L. Hennessy
ACM Transactions on Programming Languages and Systems (TOPLAS) October 1990
Volume 12 Issue 4

Global register allocation plays a major role in determining the efficacy of an optimizing compiler. Graph coloring has been used as the central paradigm for register allocation in modern compilers. A straightforward coloring approach can suffer from several shortcomings. These shortcomings are addressed in this paper by coloring the graph using a priority ordering. A natural method for dealing with the spilling emerges from this approach. The detailed algorithms for a priority-based colori ...

Results 1 - 20 of 200 **short listing**
⇦ Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page ⇨

RAM Fee History Query

Revenue Accounting and Management

Name/Number: 09608313 Total Records Found: 6

Start Date: Any Date End Date: Any Date

Accounting Date	Sequence Num.	Tran Type	Fee Code	Fee Amount	Mailroom Date	Payment Method
07/11/2000	00000103	1	101	$690.00	06/30/2000	CK
07/11/2000	00000104	1	102	$156.00	06/30/2000	CK
07/11/2000	00000105	1	103	$54.00	06/30/2000	DA 022666
07/11/2000	00000106	1	103	$792.00	06/30/2000	CK
11/02/2000	00000142	1	105	$130.00	10/30/2000	CK
11/09/2000	00000027	1	581	$40.00	10/30/2000	CK